

be converted by updating new or modified information based on old information in the original data file. The old information can also be retained without any changes.

[0065] Steps **570** and **580** operate to convert the original data file based on the last version watermark. The last version watermark can indicate additional information from the high version watermark to determine what changes to the original data file are needed. For example, suppose that version 2 of the computer application program (the high version watermark) has written the original data file. Version 2 can include a new or modified property from Version 1. However, if version 1 of the application program was the last version to write the file (the last version watermark), then it may have changed some of the data that corresponds to the new or modified properties of version 2. Accordingly, if version 2 or a later version (the active version) reads the original data file, then that version may need to convert data in the data file because the last version to write the file was version 1 (data written by version 1 may need to be updated to correspond to version 2 or the later version).

[0066] The file version watermarks can be provided on an object by object basis instead of, or as well as, on the whole file. Method **500** (and method **600** discussed below) can then be applied to an object in the original data file. Providing file version watermarks for individual objects can optimize the loading and saving methods by determining which objects in the file need to be converted or ignored. Additionally, file version watermarks can be provided on a property by property basis instead of, or as well as, on the whole file or on a whole object. Method **500** (and method **600** discussed below) can then be applied to a property in the original data file. Providing file version watermarks for individual properties can optimize the loading and saving methods by determining which properties in the file need to be converted or ignored.

[0067] Furthermore, the active version of the application program can have an active version watermark associated with each object, which can represent the versions of the application program that have revised, changed, or deleted the object. In method **500** (and method **600** discussed below), the object version watermark for a particular object can be compared to the active version, where the active version is represented by the active version watermark for that particular object in the active version. Accordingly, the present invention can optimize the determination of whether the object in the original data file needs to be updated, because the object only needs to be updated if the active version watermark is newer than the object version watermark. For example, suppose that the object version watermark for a particular object in the original data file indicates that version 1.0 was the last version to save the particular object. If active version 2.0 loads the file, it will update information in the active version that is based on the particular object, because version 2.0 is newer than version 1.0. However, if an active version watermark is provided for that particular object in the active version, where the active version watermark indicates that the particular object has not changed since version 1.0, then the active version will not update the particular object, because it has not been changed in version 2.0. Thus, the loading method can be optimized by eliminating unnecessary conversion steps.

[0068] Referring now to **FIG. 6**, an exemplary embodiment of a method **600** for saving a data file according to the

present invention will be described. Method **600** allows a modified data file to be saved so that it is compatible with an active version, a previous version, and a later version. After an original data file has been modified by the active version of the application program, the modified original data file can be saved to be compatible with the active version, previous versions, and later versions. **FIG. 6** is a flow chart depicting method **600**. Method **600** can comprise step **605** of providing a high version watermark, a low version watermark, a last version watermark, a creation version watermark, and/or an object version watermark in an original data file. In step **610**, the high version watermark in the original data file is determined. Method **600** also can determine the low version watermark, the last version watermark, the creation version watermark, and/or the object version watermark of the original data file, as shown in step **615**. In step **620**, the high version watermark is compared to the active version of the application program. Based on that comparison, the method determines how to save the modified original data file.

[0069] If the comparison of step **620** indicates that the high version watermark represents a version of the application program that is older than the active version, then the original data file corresponds to a previous version of the application program with respect to the active version. In that case, the method branches to step **625** to save the modified original data file in the format of the active version of the application program. Because the high version watermark corresponds to a previous version of the application program, the active version of the application program also can save the modified original data file in a format compatible with all previous versions. Accordingly, the method proceeds to step **630** where items in the modified original data file are converted to the format of the previous version(s) and saved. Thus, the modified original data file is saved in the formats of both the active version and the previous version(s). Old information from the previous version can be reconstructed in step **630**, allowing the file to be used by previous, active, and future versions of the application program. If reconstructing the old information becomes too large a task, then future files can be truncated by inserting a command in the file that states that the file can only be read by a certain version or future versions thereof. In step **635**, the high version watermark of the original data file is updated to correspond to the active version of the application program. The method then proceeds to step **665** where the last version to save the file also is updated for future use.

[0070] If the comparison of step **620** indicates that the high version watermark represents a version of the application program that is newer than the active version, then the original data file corresponds to a later version of the application program with respect to the active version. In that case, the method branches to step **640** where the original data file is obtained. Then, in step **645**, the original data file is overwritten with the modified original data file. Because the high version watermark corresponds to a later version of the application program, the modified original data file can contain information or data that is unknown to the active version. (For example, see step **580** in **FIG. 5**.) Accordingly, in step **645**, only known data of the original data file is overwritten with data of the modified original data file. Unknown property data is not changed. By this method, the unknown data remains available for later versions of the